Module Fifteen

Configuration Management

This module describes the concept of configuration management(CM). It discusses the TCSEC requirements for CM at the higher trust classes. It then takes a detailed look at the many different aspects of a CM system, and highlights the need for a well-documented systematic program forconducting CM.

Module Learning Objectives

This module presents material that can be read independently of theother modules. Upon completion of this module, the student should:

1. Understand what CM is.
2. Understand the TCSEC requirements for CM.
3. Be familiar with the objectives and components of a CM system.
4. Be familiar with the purpose and contents of a Configuration Management Plan (CM-Plan).

Overview

The goal of CM is to maintain control over the TCB and protect itagainst unauthorized changes that could cause protection mechanisms to malfunction or be bypassed completely. Authorized changes/updates to a system under development are inevitable, and CM ensures that these changes occur in a controlled manner so as not to adversely affect the implementation of the security policy of the TCB. Four main objectives exist for CM: identification, control, accounting, and auditing.

Configuration Identification involves identifying the design and implementation components of the TCB at a discrete point in time. The smallest portion of the system to be subject to independentconfiguration management change control procedures is identified as a configuration item (CI). A CI is a unique, identifiable subset of the system configuration. For example, if a system mail facility was not subject to much change, the entire mail facility could be one CI; conversely, if it were subject to frequent changes, each module of the mail facility (e.g., send_mail, receive_mail, edit_mail, MLS_mailbox_driver, mail_com) could be specified as a CI. When selecting the size of the CIs, the developer must balance the volume of changes associated with a single large CI against a large (potentially unmanageable)set of smaller CIs. In addition, proper configuration identification should permit theaccurate reproduction of any past TCB configuration. For examples of CIs, see the appendices in the back of any Final Evaluation Report (FER) for a current evaluated product.

Configuration Control involves the systematic evaluation,coordination, approval, or disapproval of proposed changes to the design and construction of a configuration item whose configuration has been formally approved.There are two methods for managing changes to a system: configuration enforcement and configuration review. Configuration enforcement requires that the proposed change be analyzed before it is implemented. Configuration review requires that the effects of changes be analyzed after they are implemented but before they are integrated into the system. For B2 and above systems, both checks are necessary. Configuration control should provide for constant checking and approval of a change from its inception, through implementation and test, to release. The TCSEC requires that changes to the TCB be approved, monitored, and evaluated through configuration control procedures toprovide assurance that the TCB continues to function properly.

Configuration Accounting records, stores, and reports data on the progress of the system development that is important to the configurationmanagement process. This accounting may be done manually or through the use of automated tools such as a database. The accounting should permit the production of a current configuration list, an historical change list, the original designs, and the status of change requests and their implementation, and should provide the capability to trace all changes.

Configuration Audit verifies the consistency and completeness of the accounting information; it is quality assurance of the configuration management process. Configuration audit ensures that after a change has been made to the TCB, the security features and assurances are maintained. It verifies traceability of requirements between differing levels ofspecificity , and confirms that the CI, or system, conforms to the documentation andperforms per the requirements. The validity of configuration status accounting information should be confirmed through periodic configuration audits.

TCSEC Configuration Management Requirements

CM of design documentation and source code is required by the TCSECduring development and maintenance of B2 and above computer systems. Furthermore, new releases of evaluated systems at all classes that are to be re-evaluated under RAMP must be kept under CM since the previousevaluation. Details about the CM procedures required for RAMP are covered in Module16.

All components of the system must be identified, and baselined, at the beginning of the Evaluation Phase. Systems are identified by decomposing the system into smaller subsets, or configuration items. All configuration items must be uniquely identified by a mnemonic name, a number, or some combination. As a minimum, the system's TCB, documentation, tests, and tools (including any configuration management or audit reduction tools) mustbe broken down into configuration items. The hardware, firmware, and tools used to create the TCB need only be configuration identified (except atClass A1). All non-TCB software that is shipped as part of the installation media for the trusted product must be configuration identified as well. Ofimportance here is the distinction between configuration identified and configurationitems . Since the Trusted Product Evaluation Program is only concerned with security issues, only those parts of the system that comprise the TCB or contribute to the assurance of trust need be configuration items that areconfiguration managed. A configuration item is something that may be changed under RAMP. Configuration managed means that changes to CIs have to be affected in a controlled manner (refer to [RAMP94]). The vendor will likelyexercise the same degree of control over the rest of the product, though it is notrequired for the evaluation.

At class B2 and above, the TCSEC requires that during the development and maintenance of the TCB, a CM system must maintain control of changes to the descriptive top-level specification, other design data, implementation documentation (e.g., Trusted Facility Manual, Security Features User's Guide), source code, the running version of the object code, and test fixtures and test documentation, and must assure a consistent mapping among all documentation and code. CM tools for generation and comparison of TCB versions must also be furnished.

The configuration management requirements at Class B3 are the same as Class B2. However, the additional design documentation required at B3, which must be placed under configuration management, indirectly causes a change from the B2 to the B3 CM requirements. This change is the informal DTLS to TCB mapping that is required at B3.

In addition to the CM requirements specified for Class B3, Class A1 CM requirements specify that the CM system must be in place during theentire life-cycle of the TCB. Whereas the hardware and firmware were only configuration identified for Classes B2 and B3, Class A1 CMrequirements stipulate that security-relevant hardware and firmware must now be configuration items under configuration management. Class A1 also introduces additional design documentation that must be configuration managed, including the formal top-level specification and its informal mapping to the TCB. In addition, the CM tools themselves must be maintained under strict configuration control with added procedural safeguards toensure against unauthorized modification or destruction of the master copy of the TCB.

Configuration Management Plan

A well thought out CM-Plan should be constructed to describe how CM willbe implemented in the system and the TCB. An effective CM system should be able to show what was planned to be built, what was actually built, and what modifications are currently underway. The CM-Plan should: specify the identification scheme used to identify specific versions of the system and the specific evaluation configuration; identify the players involved in the CM process (their roles and responsibilities); define the tools, techniques, and procedures used to implement the CM process; and specify any emergency procedures to deal with bugs or flaws in the product.

Owing to the shear volume of information that must be managed and the complexity of the interrelationships between documents that must be maintained, automating the process is highly desirable, though not required by the TCSEC. Some example CM systems are examined in [Brown87b] and [Heimbig88].

Relevant Trusted Product Evaluation Questionnaire Questions

2.13 OTHER ASSURANCES

Although the configuration management criteria do not appear until class B2 in the TCSEC, the questions pertainging to configuration management below are relevant to all classes because of the NSA's Ratings Maintenance Phase (RAMP) program.

C1:

1) (a) Describe the Configuration Management (CM) system inplace in terms of organizational responsibilities, procedures, and tools and techniques (automated, manual, or a combination of the two). (b) Describe the version control or other philosophy to ensurethat the elements represent a consistent system, i.e., object code represents the source code, and the design documentation accurately describes the source code. (c) If the CM system is different for some of the elements listed in question 1 in section 2.4, answer this question for each of the elements.

2) (a) When was this system placed under configuration management? (b) Provide the approximate date, as well as the life-cycle phase (e.g., design, development, testing). Answer this question for each system element so controlled (as listed in the previous question).

3) List the elements that are and are not under the Configuration Management (e.g., hardware, firmware, formal security policy model, FTLS, DTLS, design data and documentation, source code, object code, test plans, Security Features User's Guide, Trusted Facilities Manual).

4) Describe the protection mechanisms in place to safeguard the CM elements.

Required Readings

TCSEC85 National Computer Security Center, *Department of Defense Trusted Computer Security Evaluation Criteria*, DoD 5200.28-STD, December 1985.

Sections 3.2.3.2.3, 3.3.3.2.3, and 4.1.3.2.3 contain the CM requirements, which are summarized on pages 96-97.

INTERP94 National Computer Security Center, *The Interpreted TCSEC Requirements*, (quarterly).

The following Interpretations are relevant to CM.

I-0285	CM comparison source or object?
C1-CI-02-85	Audit

Brown87b Brown, R.L., *Configuration Management for Development of a Secure Computer System*, ATR-88(3777-12)-1, The Aerospace Corporation, December 1987.

This is a draft guideline prepared by the Aerospace Corporation on CM for operating system software and computer hardware that describes the minimum CM effort required by the TCSEC for classes B2 through A1 and recommends additional requirements.

CM88 National Computer Security Center, *A Guide to Understanding Configuration Management in Trusted Systems*, NCSC-TG-006, Version 1, 28 March 1988.

This document provides guidance on TCSEC CM requirements and discusses issues involved in implementing CM in the development and life-cycle of a trusted system.

Supplemental Readings

Brown87a Brown, R.L., "Specification for a Canonical Configuration Accounting Tool," *Proceedings of the 10th National Computer Security Conference*, pp. 84-90, September 1987.

This paper overviews two commercial automated document control facilities meeting the needs of configuration accounting: Unix SCCS and VAX DEC/CMS. It presents a draft guideline for a canonical Text and Code Control System that may be used as an aid to evaluating other configuration accounting systems to be used in the development of secure systems.

Heimbig88 Heimbigner, D. and Krane, S., "A Graph Transform Model for Configuration Management Environments," *Proceedings of the ACM SIGSOFT/SIGPLAN Software Engineering Symposium on Practical Software Development Environments*, November 1988.

This paper describes a model for CM that is patterned after a compiler that in multiple phases transforms a program into an executable. The transformational approach is used to model and compare four existing CM systems.

RAMP94 National Computer Security Center, *Rating Maintenance Phase: Program Document*, Draft, Version 2, 1 March 1994.

Other Readings

ACM89 *Proceedings of the 2nd International Workshop on Software Configuration Management*, Princeton, New Jersey, ACM Press, 24 October 1989.

A collection of 30 papers that provide a diverse set of current viewpoints on topics related to supporting and controlling the evolution of large software systems.

Cohen88 Cohen, E., Soni, D., Gluecker, R., Hasling, W., Schwanke, R., and Wagner, M., "Version Management in Gypsy," *Proceedings of the ACM SIGSOFT/SIGPLAN Software Engineering Symposium on Practical Software Development Environments*, November 1988.

Mahler88 Mahler, A. and Lampen, A., "An Integrated Toolset for Engineering Software Configurations," *Proceedings of the ACM SIGSOFT/SIGPLAN Software Engineering Symposium on Practical Software Development Environments*, November 1988.